2007

An Interactive Constraint-Based Approach to Sudoku

Christopher Reeson
University of Nebraska-Lincoln, creeson@cse.unl.edu

Kai-Chen Huang
University of Southern California, kaichenh@usc.edu

Ken Bayer
University of Nebraska-Lincoln, kbayer@cse.unl.edu

Berthe Y. Choueiry
University of Nebraska-Lincoln, choueiry@cse.unl.edu

An Interactive Constraint-Based Approach to Sudoku

Christopher G. Reeson[1] · Kai-Chen Huang[2] · Kenneth M. Bayer[1] · Berthe Y. Choueiry[1,2]

[1]Constraint Systems Laboratory, University of Nebraska-Lincoln · [2]Information Sciences Institute, University of Southern California

SOLVER: sudoku.unl.edu/Solver • **CONSTRUCTOR**: sudoku.unl.edu/Constructor

Goals

- To the public:
  - Illustrate the power of CP
- For the research:
  - Investigate the use of CP to interactively support & guide human players
- For education:
  - Teach basic and advanced CP techniques
- For insight into human reasoning:
  - Understand how it differs from algorithmic approaches

Sudoku

A Sudoku…

- is well posed if it has a single solution
- is minimal if removing any 'given' yields more than one solution
- is symmetric if the filled cells on the grid exhibit some axial symmetry

							1	3
			7				6	
			5		9			
			4			9		
1		6						
						2		
7	4						5	
	8					4		
				1				

#15 on Royle's web site[1]

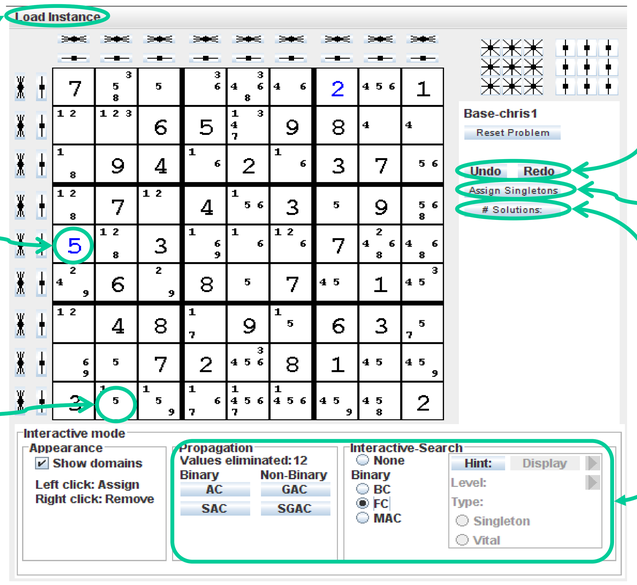
SOLVER

SOLVER is a Java applet that uses CP techniques to support the user to play Sudoku. The player can…

…load an instance from the online library

…assign values to cells and play the game without aid

…display the remaining values in the domain of each cell



…undo/redo any action

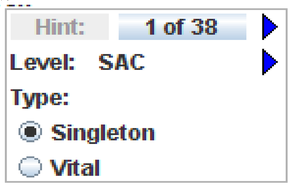
…assign all cells whose domains have a single value

…check the number of solutions left

…apply a variety of CP techniques

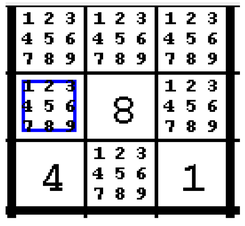
SOLVER displays the number of hints. The user can iterate through them.

The user can choose the type of hint & the level of consistency:

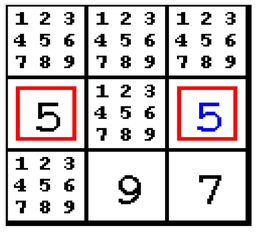


- 2 types of hints: Singleton & Vital
- 8 levels: FC, AC, Single GAC, GAC, Single SAC, SAC, Single SGAC, SGAC

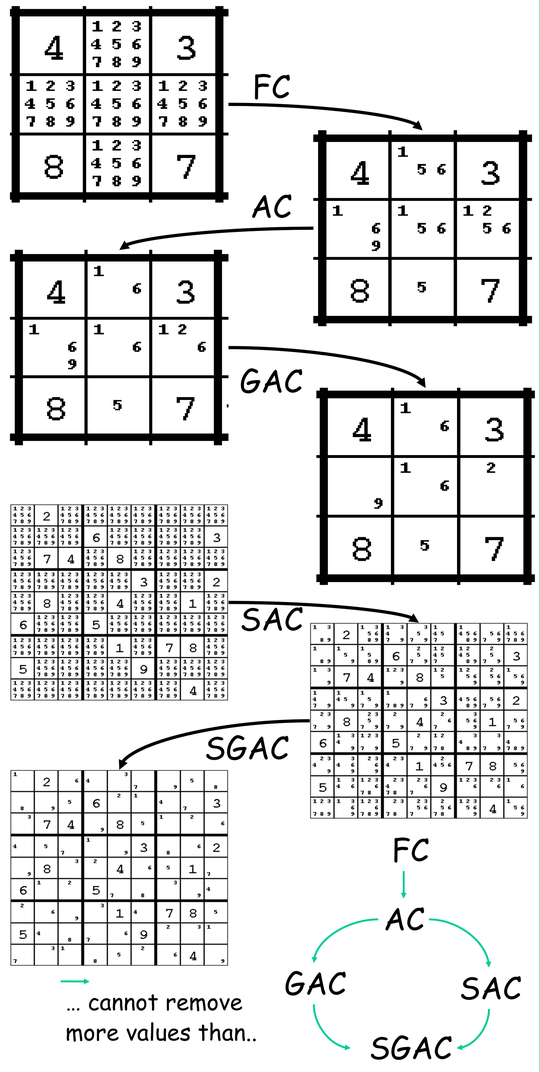
HINT highlights the cell that the player needs to think about..



SOLVER detects errors and highlights the variables in the broken constraint…

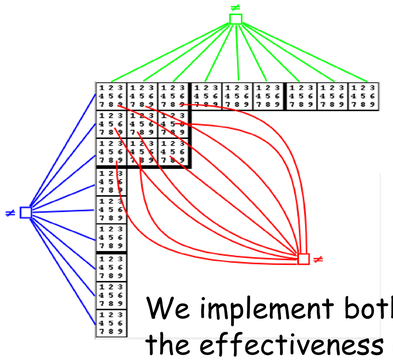


Constraint Propagation

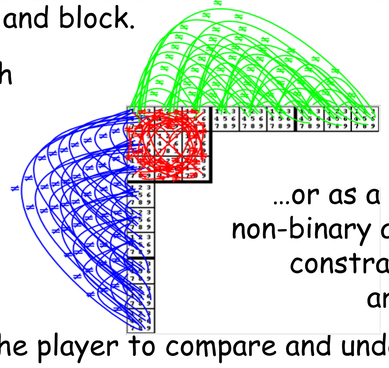


CSP Models

Each cell is a variable whose domain is set of numbers 1…9. The constraints are 'all-different' constraints on each row, column, and block.



We can model each all-diff constraint either as a set of binary mutex constraints…

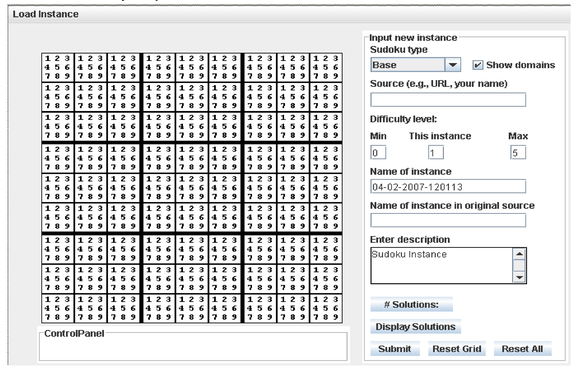


…or as a single non-binary alldiff constraint of arity 9.

We implement both models to allow the player to compare and understand the effectiveness of constraint propagation operating on each model.

CONSTRUCTOR

- Supports entering & storing instances
- Counts & displays all solutions



Conjectures:

1. SGAC solves well-posed 9x9 Sudoku
2. SGAC ≡ relation (1,2)consistency

Observations: Human players…

… can do GAC after some training

… cannot mentally do shaving (i.e., SAC, SGAC)

… solve Sudoku using convoluted patterns with imaginative names. Often, several patterns correspond to the same propagation mechanism

Support: UCARE grant awarded to Chris Reeson & CAREER Award #0133568 from the National Science Foundation.

References

1. Gordon Royle. Minimum Sudoku. people.csse.uwa.edu.au/gordon/sudokumin.php, 2005
2. Helmut Simonis. Sudoku as a Constraint Problem. Workshop on Modeling and Refomulating CSPs, 2005

An Interactive Constraint-Based Approach to Sudoku

Christopher G. Reeson[1] · Kai-Chen Huang[2] · Kenneth M. Bayer[1] · Berthe Y. Choueiry[1,2]

[1]Constraint Systems Laboratory, University of Nebraska-Lincoln · [2]Information Sciences Institute, University of Southern California

SOLVER: sudoku.unl.edu/Solver • **CONSTRUCTOR**: sudoku.unl.edu/Constructor

Goals

- To the public:
  - Illustrate the power of CP
- For the research:
  - Investigate the use of CP to interactively support & guide human players
- For education:
  - Teach basic and advanced CP techniques
- For insight in human reasoning:
  - Understand how it differs from algorithmic approaches

Sudoku

A Sudoku…

- is well posed if it has a single solution
- is minimal if removing any 'given' yields more than one solution
- is symmetric if the filled cells on the grid exhibit some axial symmetry

							1	3
			7				6	
			5		9			
			4			9		
1		6						
						2		
7	4						5	
	8					4		
				1				

#15 on Royle's web site[1]

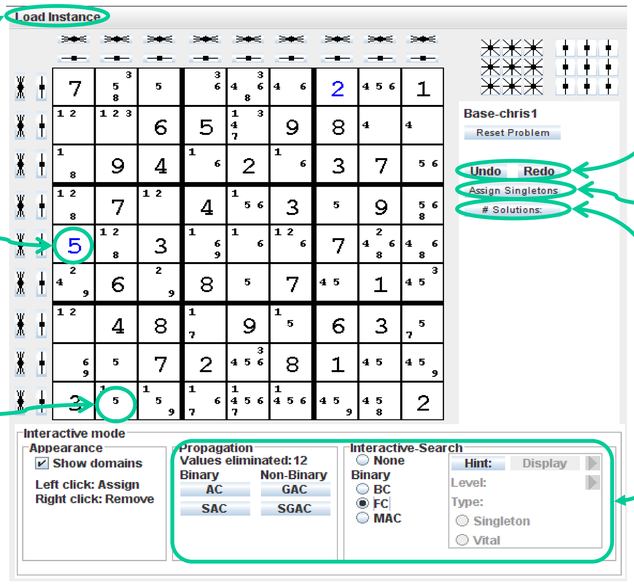
SOLVER

SOLVER is a Java applet that uses CP techniques to support the user to play Sudoku. The player can…

…load an instance from the online library

…assign values to cells and play the game without aid

…display the remaining values in the domain of each cell



…undo/redo any action

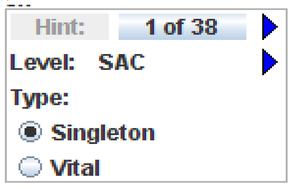
…assign all cells whose domains have a single value

…check the number of solutions left

…apply a variety of CP techniques

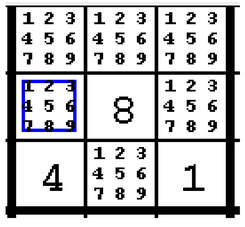
SOLVER displays the number of hints. The user can iterate through them.

The user can choose the type of hint & the level of consistency:

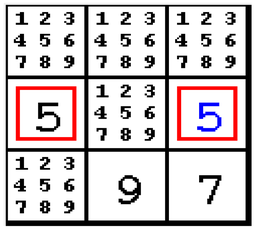


- 2 types of hints: Singleton & Vital
- 8 levels: FC, AC, Single GAC, GAC, Single SAC, SAC, Single SGAC, SGAC

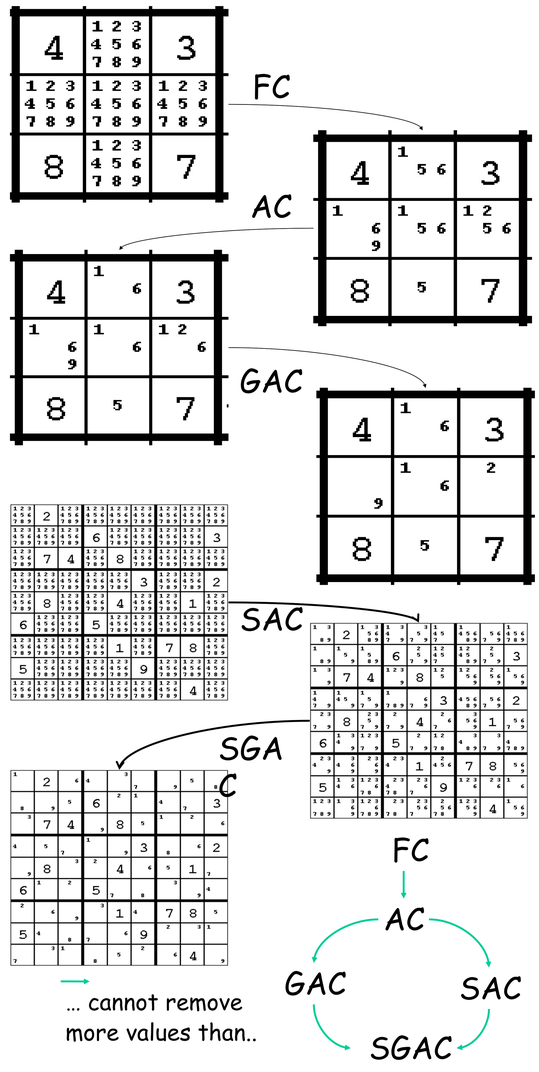
HINT highlights the cell that the player needs to think about..



SOLVER detects errors and highlights the variables in the broken constraint…



Constraint Propagation



Conjectures:

1. SGAC solves well-posed 9x9 Sudoku
2. SGAC ≡ relation (1,2)consistency

Observations: Human players…

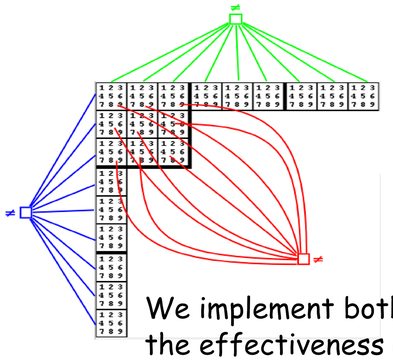
… can do GAC after some training

… cannot mentally do shaving (i.e., SAC, SGAC)

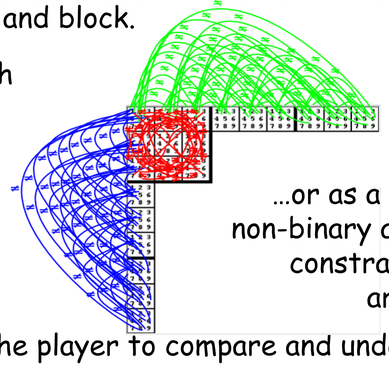
… solve Sudoku using convoluted patterns with imaginative names. Often, several patterns correspond to the same propagation mechanism

CSP Models

Each cell is a variable whose domain is set of numbers 1…9. The constraints are 'all-different' constraints on each row, column, and block.



We can model each all-diff constraint either as a set of binary mutex constraints…

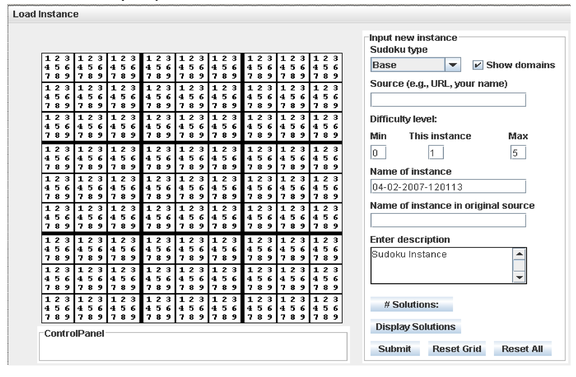


…or as a single non-binary alldiff constraint of arity 9.

We implement both models to allow the player to compare and understand the effectiveness of constraint propagation operating on each model.

CONSTRUCTOR

- Supports entering & storing instances
- Counts & displays all solutions



Support: UCARE grant awarded to Chris Reeson & CAREER Award #0133568 from the National Science Foundation.

References

1. Gordon Royle. Minimum Sudoku. people.csse.uwa.edu.au/gordon/sudokumin.php, 2005
2. Helmut Simonis. Sudoku as a Constraint Problem. Workshop on Modeling and Reformulating CSPs, 2005